BytesToUnicode

Be careful with size parameters

Sean Barnum, Cigital, Inc. [vita[1]]

2007-02-27

Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 5242 bytes

Attack Category	• Malicious Input
Vulnerability Categories	• Multibyte Character • Buffer Overflow
Software Context	• String Formatting • String Parsing • String Conversion MACROS • String Management
Location	• GB18030.h
Description	Using the BytesToUnicode() function incorrectly can create a potential for buffer overflow. The BytesToUnicode() function converts bytes representing characters in the GB18030 encoding (the official character set of the Peoples Republic of China) to Unicode characters. GB18030 encodes characters as 1, 2 or 4 bytes, whereas Unicode (UCS-2) characters are each 2 bytes. Arguments to BytesToUnicode() include lpMultiByteStr, the number of bytes in the input text, and lpWideCharStr, the number of characters that the output buffer can hold. The presence of arguments specifying sizes in two different ways can lead to erroneous usage. If a programmer erroneously specifies the size of the output buffer in bytes instead of characters, then BytesToUnicode() will expect the buffer to be larger than it really is, and a buffer overrun can easily occur. Note that usage of MultiByteToWide() is preferred over BytesToUnicode(), but that function is subject to the same security issue.
APIs	**FunctionName** / **Comments** BytesToUnicode
Method of Attack	If buffer size is specified as number of bytes instead of number of characters then a buffer overflow may

1. http://buildsecurityin.us-cert.gov/bsi/about_us/authors/35-BSI.html (Barnum, Sean)

<table>
<tr><td></td><td>occur, resulting in unpredictable behavior. If an attacker is able to control the text to be converted, then he or she can mount a buffer overflow attack.</td></tr>
<tr><td>Exception Criteria</td><td>If BytesToUnicode() is called with its cchWideChar argument correctly set to the output buffer size in characters (not bytes), then this issue does not apply.</td></tr>
<tr><td>Solutions</td><td><table>
<tr><th>Solution Applicability</th><th>Solution Description</th><th>Solution Efficacy</th></tr>
<tr><td>Always applicable.</td><td>Specify the output buffer size in characters rather than in bytes.</td><td>Entirely effective.</td></tr>
</table></td></tr>
<tr><td>Signature Details</td><td>Final argument to BytesToUnicode(), lpWideCharStr, is a number of bytes rather than a number of characters.</td></tr>
<tr><td>Examples of Incorrect Code</td><td><pre>LPWSTR resultBuffer =
(LPWSTR)malloc(BUFFER_BYTE_COUNT);
charactersConverted =
BytesToUnicode(lpMultiByteStr,
cchMultiByte, pcchLeftOverBytes,
resultBuffer, BUFFER_BYTE_COUNT);</pre></td></tr>
<tr><td>Examples of Corrected Code</td><td><pre>LPWSTR resultBuffer =
(LPWSTR)malloc(BUFFER_CHARACTER_COUNT
* sizeof(WCHAR));
charactersConverted =
BytesToUnicode(lpMultiByteStr,
cchMultiByte, pcchLeftOverBytes,
resultBuffer,
BUFFER_CHARACTER_COUNT);</pre></td></tr>
<tr><td>Source Reference</td><td>MSDN BytesToUnicode[2]</td></tr>
<tr><td>Recommended Resources</td><td>BytesToUnicode[3]
MultibyteToWideChar description[4]</td></tr>
<tr><td>Discriminant Set</td><td><table>
<tr><td>Operating System</td><td>Windows</td></tr>
<tr><td>Language</td><td>C
C++</td></tr>
</table></td></tr>
</table>

Cigital, Inc. Copyright

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.

1. mailto:copyright@cigital.com